|  |  |
| --- | --- |
|  | УТВЕРЖДАЮ |
|  | Руководитель Учебной практики  \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Долженкова М. Л.  «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_ г.  М.П. |

ТЕХНИЧЕСКАЯ ДОКУМЕНТАЦИЯ

Информационной системы по учету перевозок пассажиров на электропоездах

|  |
| --- |
| СОГЛАСОВАНО |
| Колледж ВятГУ  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_ г.  М.П. |
|  |
|  |

2024

1. **Уточнение структуры данных:**

Набор сущностей с их свойствами для ИС по учету перевозок пассажиров на электропоездах представлен в таблице 1.

Таблица 1 – Сущности и атрибуты данных

|  |  |
| --- | --- |
| Сущность | Свойство |
| Поезда | Номер поезда |
| Вид |
| Количество вагонов |
| Маршруты | Номер маршрута |
| Откуда |
| Куда |
| Номер поезда |
| Время отправления |
| Время прибытия |
| Цена билета |
| Пассажиры | Номер пассажира |
| Пасспорт |
| Фамилия |
| Имя |
| Номер маршрута |
| Номер поездки |

Контекстная диаграмма информационной системы по учету перевозок пассажиров на электропоездах представляет собой обобщенное изображение взаимодействия приложения с внешней средой.

Контекстная диаграмма представлена на рисунке 1.



Рисунок 1 – Контекстная диаграмма информационной системы по учету перевозок пассажиров на электропоездах

На данной диаграмме изображен блок, обозначающий учет перевозок пассажиров на электропоездах.

Внешней входящей информацией является:

* Запросы пассажира.

Управляющим воздействием будет являться:

* ГОСТ Р 58855-2020 Услуги на железнодорожном транспорте. Качество услуг в области грузовых перевозок. Термины и определения (дата введения 01.12.2020) - Установленные в настоящем стандарте термины расположены в систематизированном порядке, отражающем систему понятий, связанных с обеспечением качества услуг в сфере перевозок на железнодорожном транспорте.
* ГОСТ Р 58171— 2018 Услуги на железнодорожном транспорте. Требования к обслуживанию пассажиров на вокзальных комплексах – настоящий стандарт распространяется на процессы, связанные с обслуживанием пассажиров и посетителей на территории железнодорожных вокзальных комплексов.

Механизмом управления является:

* База данных.
* Администратор.

Внешней входящей информацией является:

* Оформление билета.

Далее будут представлены декомпозиции контекстной диаграммы. Они представляют контекстную диаграмму разложением на набор составляющих ее более простых функций.

На рисунке 2 представлена декомпозиция процесса «Работа с поездами».

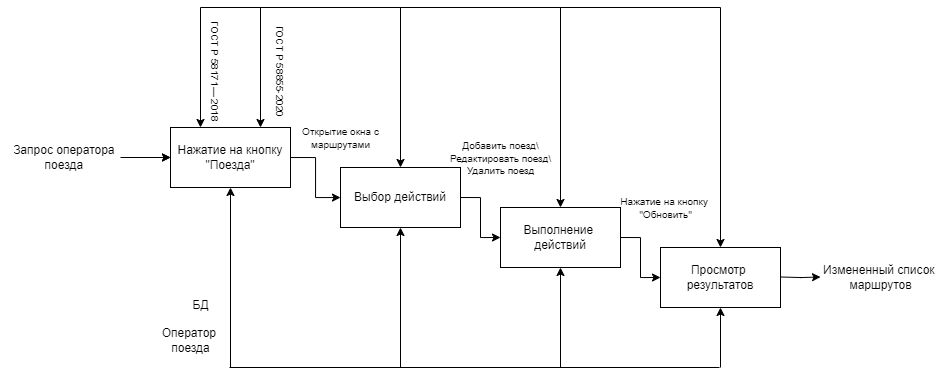


Рисунок 2 – Диаграмма IDEF декомпозиция.

Данная декомпозиция состоит процессов таких, как нажатие на кнопку «Поезда», выбор действий, выполнений действий, просмотр результатов. Входящей информаций будет являться запрос оператора поезда. Механизмом управления является оператор поезда и база данных. Управляющим воздействием будут являться ГОСТ Р 58855-2020 и ГОСТ Р 58171— 2018. Оператор поезда в главном окне нажимает на кнопку «Поезда» и дальше он может выполнить различные действия. Если поезда нету, он может добавить новый поезд, введя данные для этого поезда. Если запись поезда требует корректировок, оператор поезда может отредактировать данные поезда. Также он может удалить его. После выполненных действий он может нажать на кнопку «Обновить». В результате новые или измененные данные поезда будут отображены.

На рисунке 3 представлена декомпозиция процесса работа «Работа с маршрутами».

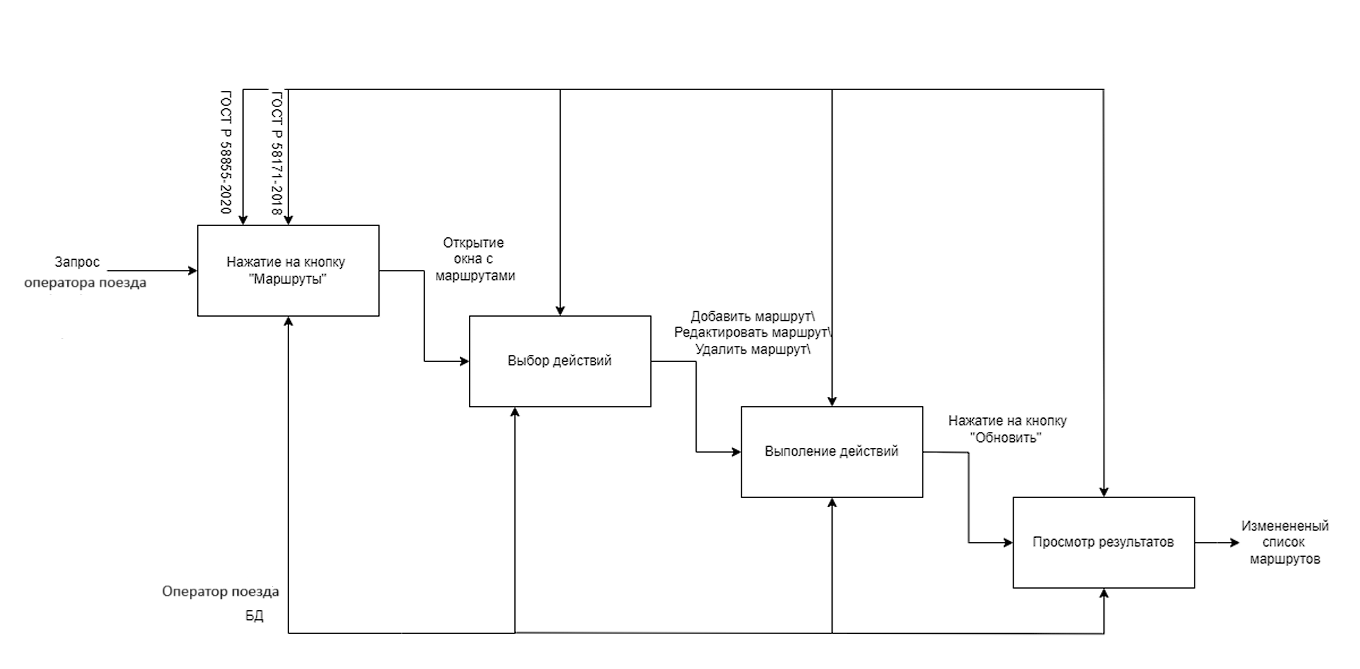


Рисунок 3 – декомпозиция процесса «Работа с маршрутами»

Данная декомпозиция состоит процессов таких, как нажатие на кнопку «Маршруты», выбор действий, выполнений действий, просмотр результатов. Входящей информаций будет являться запрос оператора поезда. Механизмом управления является оператор поезда и база данных. Управляющим воздействием будут являться ГОСТ Р 58855-2020 и ГОСТ Р 58171— 2018. Оператор поезда в главном окне нажимает на кнопку «Маршруты» и дальше он может выполнить различные действия. При добавлении нового маршрута он вводит данные о маршруте и добавляет поезд, который будет прикреплен к данному маршруту. Если запись маршрута требует изменений, оператор поезда может отредактировать данные маршрута. Также он может удалить его. После выполненных действий он может нажать на кнопку «Обновить». В результате новые или измененные данные маршрута будут отображены.

На рисунке 4 представлена декомпозиция процесса «Регистрация пассажира».

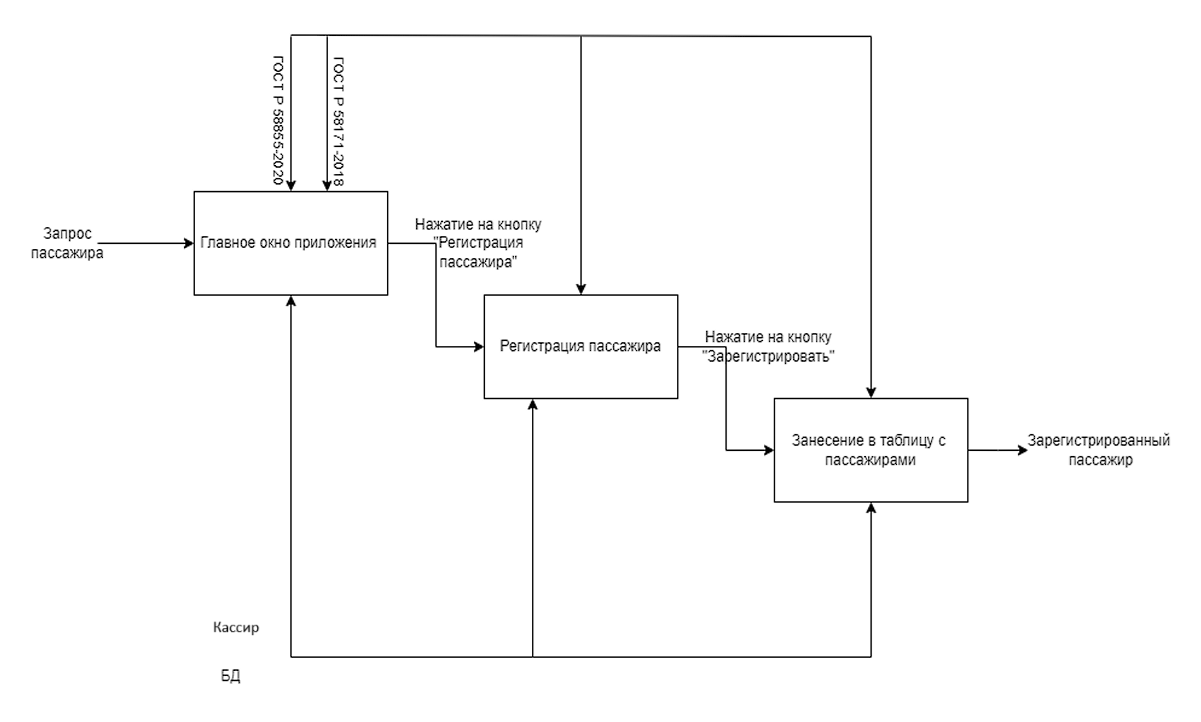


Рисунок 4 – декомпозиция процесса «Регистрация пассажира»

Декомпозиция состоит процессов таких, как главное окно приложения, регистрация пассажира, занесение результатов в окно к пассажирам. Входящей информаций будет являться запрос пассажира. Механизмом управления является кассир и база данных. Управляющим воздействием будут являться ГОСТ Р 58855-2020 и ГОСТ Р 58171— 2018. Кассир получает запрос пассажира и главном окне приложения нажимает на кнопку «Регистрация пассажира». Затем он вводит необходимые данные для регистрации пассажира. Далее он нажимает на кнопку «Зарегистрировать» и в результате данные пассажира заносятся в окно к пассажирам.

На рисунке 5 представлена декомпозиция процесса «Печать билета».

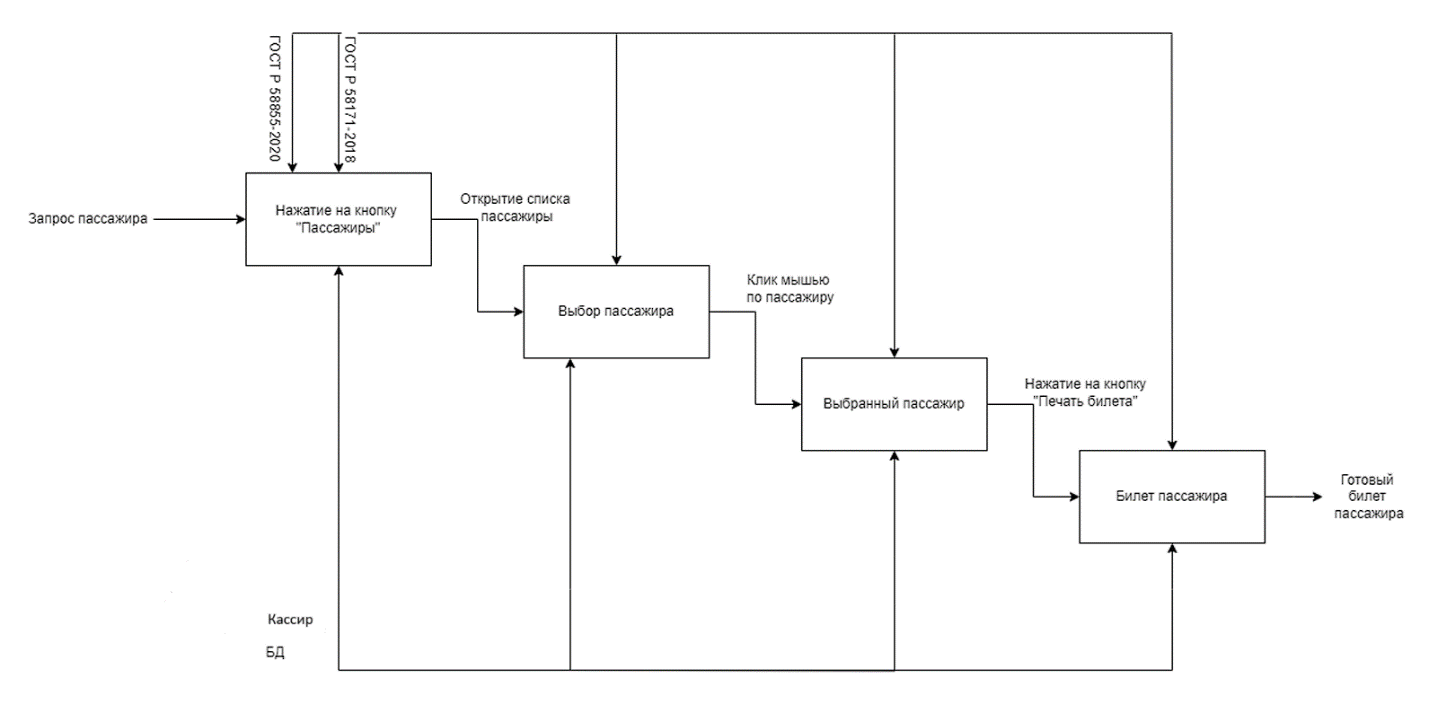


Рисунок 5 – декомпозиция процесса «Печать билета»

Входящей информацией будет также запрос пассажира. Управлять процессом будет кассир с базой данных. Управляющим воздействием являются ГОСТ Р 58855-2020 и ГОСТ Р 58171— 2018. Кассир нажимает на кнопку «Пассажиры». Далее он выбирает пассажира, от которого получил запрос. После выбора пассажира, нажимает на кнопку «Печать билета». В результате открывается окно с билетом, который содержит необходимые данные.

На рисунке 6 представлена декомпозиция процесса «Статистика».

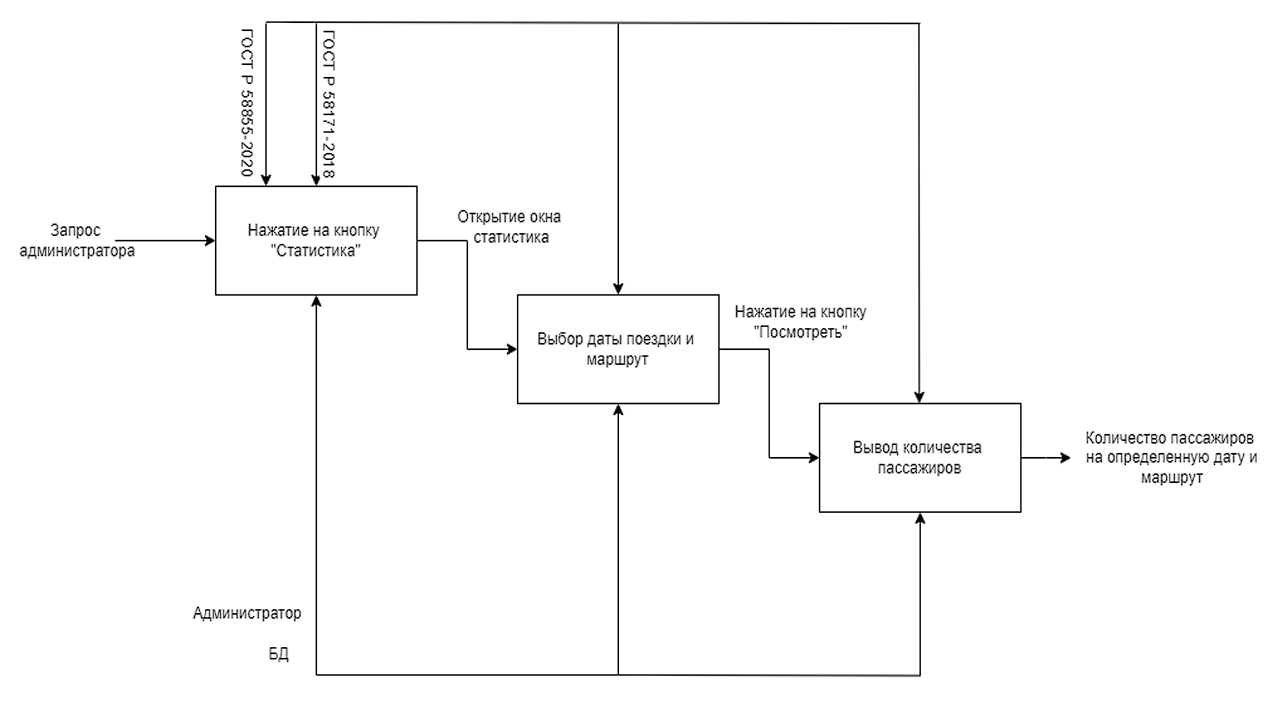


Рисунок 6 – декомпозиция процесса «Статистика»

Входящей информацией и управлять будет администратор. Управляющим воздействием являются ГОСТ Р 58855-2020 и ГОСТ Р 58171— 2018. Администратор нажимает на кнопку «Статистика». В открывшемся окне он выбирает дату поездки и маршрут. Далее он нажимает на кнопку «Посмотреть». В результате будут показаны количество и сами пассажиры, которые ездили в выбранную дату на выбранном маршруте.

На рисунке 7 представлена декомпозиция процесса «История поездок».

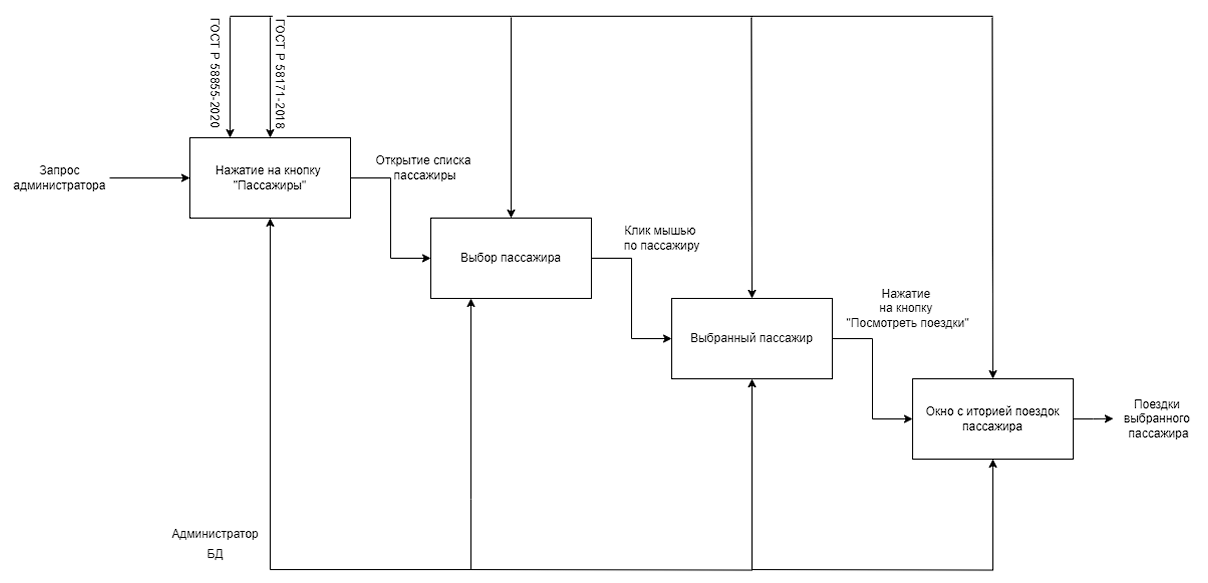


Рисунок 7 – декомпозиция процесса «История поездок»

Входящей информацией и управляющим процессом является администратор. Управляющим воздействием являются ГОСТ Р 58855-2020 и ГОСТ Р 58171— 2018. В главном окне приложения администратор нажимает на кнопку «Пассажиры». В открывшемся окне он выбирает пассажира. После выбора пассажира он нажимает на кнопку «Посмотреть поездки». В результате в окне будут показаны все поездки выбранного пассажира.

У информационной системы по учету перевозок пассажиров на электропоездах база данных изображена в логической ER диаграмме и физической ER диаграмме. Они изображены на рисунках 8 и 9.



Рисунок 3 – логическая ER-диаграмма.

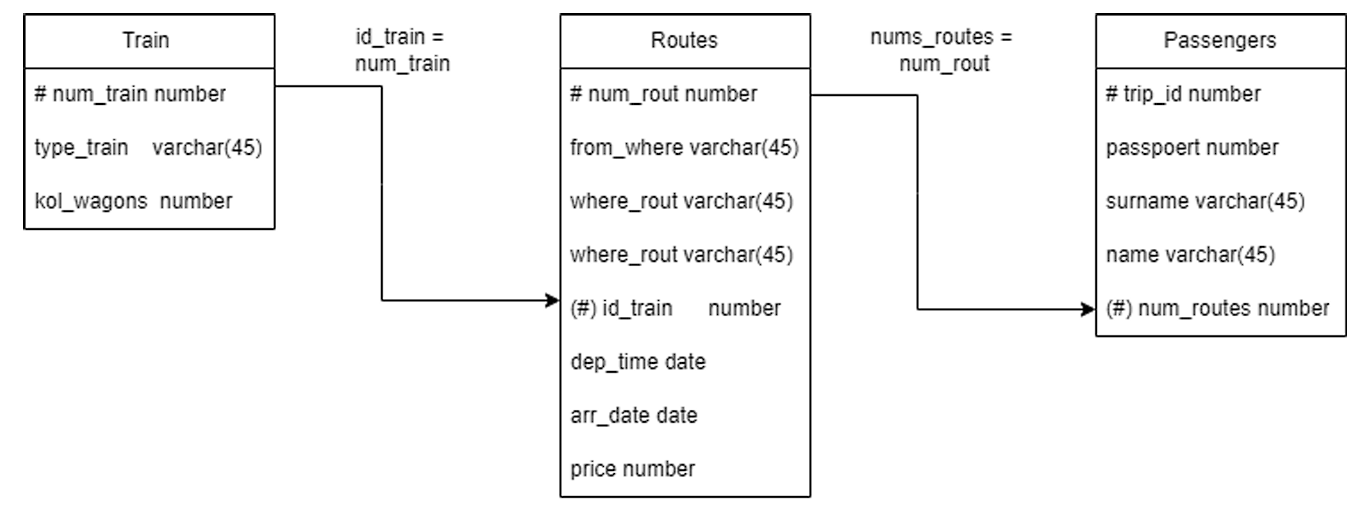


Рисунок 4 – физическая ER-диаграмма.

Связи между сущностями:

* «Маршруты» - «Поезда»: один ко многим, на одном маршруте может быть несколько поездов в разное время, но один поезд не может быть на нескольких маршрутах.
* «Пассажиры» - «Маршруты»: один ко многим, пассажир может заказать билет на несколько маршрутов, но один маршрут может быть прикреплен только к одному пассажиру. Например: Пассажир заказывает билет туда и обратно. В итоге получается 2 разных маршрута.

Целостность сущностей:

* Внешний ключ Номер поезда из таблицы Маршруты связан с первичным ключом Номер поезда из таблицы Поезда.
* Внешний ключ Номер поездки из таблицы Пассажиры связан с первичным ключом Номер маршрута из таблицы Маршруты.

1. **Формы представления данных**

На рисунке 5 представлен прототип экранной формы входа в систему.

Рисунок 5 - Прототип экранной формы «Вход в систему»

На прототипе присутствует: название системы, кнопка «Регистрация пассажира», кнопка «Маршруты», кнопка «Поезда», кнопка «Статистика», кнопка «Пассажиры».

На рисунке 6 представлен прототип экранной формы регистрации пассажира.

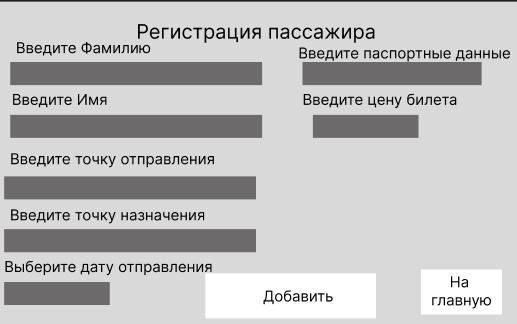


Рисунок 6 - прототип экранной формы «Регистрация пассажира»

На прототипе присутствуют кнопка «Добавить», кнопка «На главную», поля для ввода фамилии, имени, паспорта, точки отправления и прибытия, даты отправления, цены билета.

На рисунке 7 представлен прототип экрнанной формы «Пассажиры».

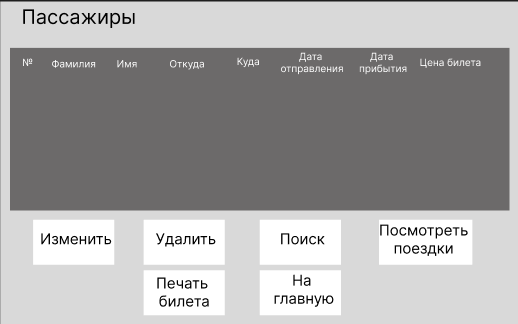


Рисунок 8 – прототип экранной формы «Пассажиры»

На прототипе присутствуют название экранной формы и информация о пассажирах. Также присутствуют кнопки «Изменить», «Удалить», «Поиск, «Посмотреть поездки», «Печать билета» и «На главную».

На рисунке 9 представлен прототип экранной формы «Маршруты».

Рисунок 9 – прототип экранной формы «Маршруты»

На прототипе присутствуют информация о маршрутах, название экранной формы. Также присутвуют кнопки «Добавить», «Изменить», «Удалить», «Поиск», «На главную».

На рисунке 10 представлен прототип экранной формы «Добавление маршрута»

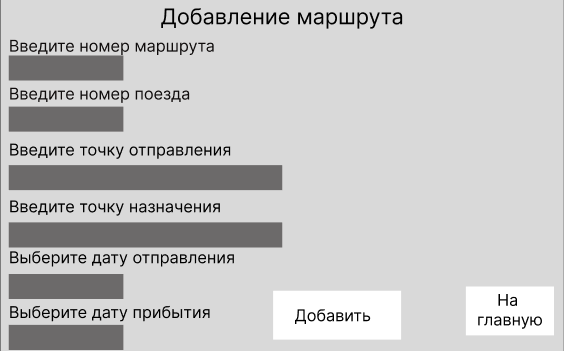


Рисунок 10 – прототип экранной формы «Добавление пассажира»

На прототипе присутствуют кнопки «Добавить» и «На главную». Также присутвуют поля для ввода номера маршрута, номера поезда, точки отправления, точки прибытия, точки назначения, даты отправления даты прибытия.

На рисунке 11 представлен прототип экранной формы «Поезда».



Рисунок 11 – прототип экранной формы «Поезда»

На прототипе присутствуют название экранной формы и информация о поездах. Также присутствуют кнопки «Добавить», «Изменить», «Удалить», «Поиск», «На главную»,

На рисунке 11 представлен прототип экранной формы «Добавление поезда».



Рисунок 11 – прототип экранной формы «Добавление поезда».

На прототипе присутствуют название экранной формы, поля для ввода номера поезда, вида и типа поезда. Также присутствуют кнопки «Добавить» и «На главную».

На рисунке 11 присутствует прототип экранной формы «Статистика».

Рисунок 11 – Прототип экранной формы «Статистика»

На прототипе присутствуют: название экранной формы, поля для выбора времени (за месяц, за день, за год), для выбора маршрута, кнопка «Посмотреть», кнопка «На главную».

На рисунке 12 присутствует прототип экранной формы «Билет».

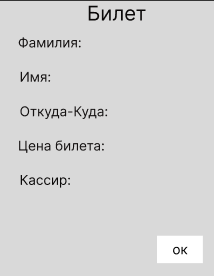


Рисунок 12- Прототип экранной формы «Билет»

На прототипе присутствуют название экранной надписи «Билет», «Имя», «Откуда-Куда», «Цена билета», «Кассир», кнопка «Ок».

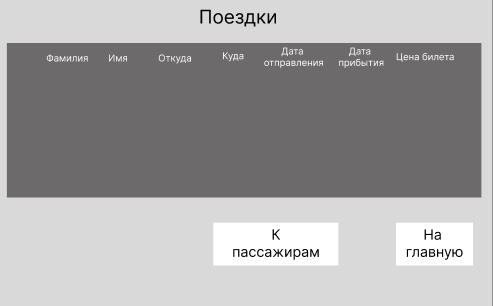
На рисунке 13 представлен прототип экранной формы «Поездки».

Рисунок 13 – прототип экранной формы «Поездки»

На прототипе присутствуют информация о поездках данного пассажира, кнопка «К пассажирам», «На главную».

1. **Разработка алгоритма решения задачи:**

Для понимания как реагирует система были составлены пользовательские сценарии.

|  |  |  |
| --- | --- | --- |
| **№** | **Пользователь** | **Система** |
| 1 | Опеартор поездов создает новый маршрут | Система создает новый маршрут |
| 2 | Оператор поездов редактирует маршрут | Система позовляет отредактировать уже созданный маршрут |
| 3 | Оператор поездов удаляет маршрут | Система позволяет удалить маршрут, если к нему не прикреплен поезд |
| 4 | Оператор поездов Добавляет поезд | Система добавляет поезд в базу данных |
| 5 | Оператор поездов Редактирует поезд | Система позволяет отредактировать добавленный поезд |
| 6 | Оператор поездов Удаляет поезд | Система позволяет удалить поезд, если он не прикреплен к маршруту |
| 7 | Оператор поездов Поиск маршрута | Система позволяет выполнять поиск маршрута, по цене билета, точки прибытия и отправления, а также датам |
| 8 | Оператор поездов Поиск поезда | Система позволяет выполнять поиск поезда, по типу и количеству вагонов |
| 9 | Кассир Регистрирует пассажира на поезд | Система добавляет нового пассажира, в котором указан паспорт, Фамилия, Имя, номер поездки, маршрут |
| 10 | Кассир и Администратор системы Просматривают поездки пассажира | Система позволяет просматривать поездки пассажира по его паспорту |
| 11 | Кассир Печатает билет | Система позволяет печать билет для пассажира |
| 12 | Адинистратор системы Просматривает статистику всех пассажиров | Система позволяет просматривать пассажиров за период времени на определенном маршруте |
| 13 | Администратор системы и Кассир Редактирует данные о пассажире | Система позволяет отредактировать данные добавленного пассажира |
| 14 | Администратор системы Удаляет данные о пассажире | Система позволяет удалить пассажира |

Для лучшего понимания алгоритма «Система создает новый маршрут» была создана схема алгоритма, которая изображена на рисунке 14.

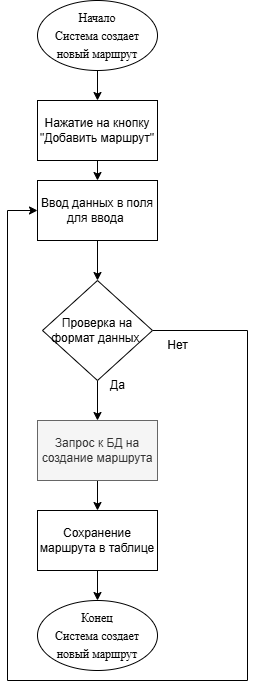


Рисунок 14 – алгоритм «Система создает новый маршрут»

Для лучшего понимания алгоритма «Система позовляет отредактировать уже созданный маршрут» была создана схема алгоритма, которая изображена на рисунке 15.



Рисунок 15 – алгоритм «Система позовляет отредактировать уже созданный маршрут»

Для лучшего понимания алгоритма «Система позволяет удалить маршрут, если к нему не прикреплен поезд» была создана схема алгоритма, которая изображена на рисунке 16.

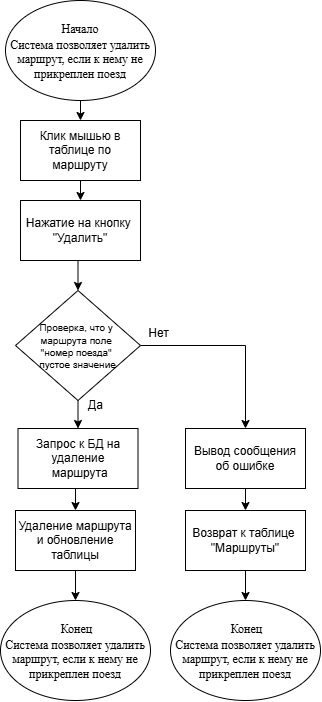


Рисунок 16 – алгоритм «Система позволяет удалить маршрут, если к нему не прикреплен поезд»

Для лучшего понимания алгоритма «Система добавляет поезд, в который входит номер поезда, тип поезда, количество вагонов» была создана схема алгоритма, которая изображена на рисунке 17.

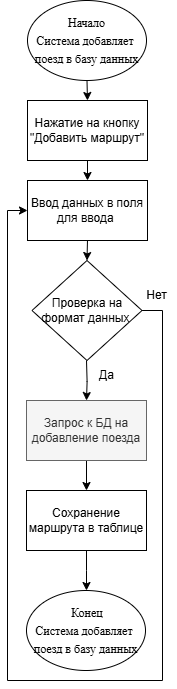


Рисунок 17 – алгоритм «Система добавляет поезд, в который входит номер поезда, тип поезда, количество вагонов»

Для лучшего понимания алгоритма «Система позволяет отредактировать добавленный поезд» была создана схема алгоритма, которая изображена на рисунке 18.



Рисунок 18 – алгоритм «Система позволяет отредактировать добавленный поезд»

Для лучшего понимания алгоритма «Система позволяет выполнять поиск маршрута, по цене билета, точки прибытия и отправления, а также датам» была создана схема алгоритма, которая изображена на рисунке 19.



Рисунок 19 – алгоритм «Система позволяет выполнять поиск маршрута, по цене билета, точки прибытия и отправления, а также датам»

Для лучшего понимания алгоритма «Система позволяет удалить поезд, если он не прикреплен к маршруту» была создана схема алгоритма, которая изображена на рисунке 20.

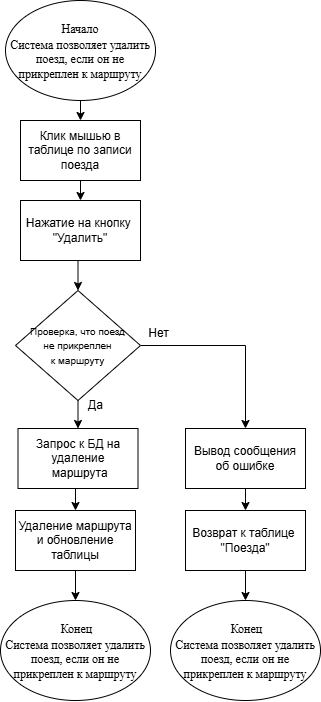


Рисунок 20 – алгоритм «Система позволяет удалить поезд, если он не прикреплен к маршруту»

Для лучшего понимания алгоритма «Система позволяет выполнять поиск поезда, по типу и количеству вагонов» была создана схема алгоритма, которая изображена на рисунке 21.



Рисунок 21 – алгоритм «Система позволяет выполнять поиск поезда, по типу и количеству вагонов»

Для лучшего понимания алгоритма «Система добавляет нового пассажира, в котором указан паспорт, Фамилия, Имя, номер поездки, маршрут» была создана схема алгоритма, которая изображена на рисунке 22.

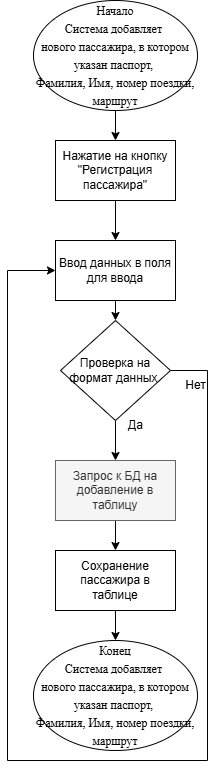


Рисунок 22 – алгоритм «Система добавляет нового пассажира, в котором указан паспорт, Фамилия, Имя, номер поездки, маршрут»

Для лучшего понимания алгоритма «Система позволяет просматривать поездки пассажира по его паспорту» была создана схема алгоритма, которая изображена на рисунке 23.

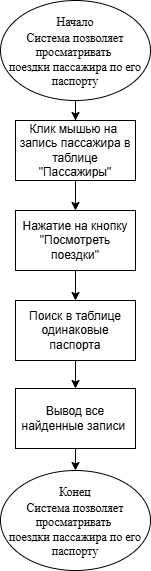


Рисунок 23 – алгоритм «Система позволяет просматривать поездки пассажира по его паспорту»

Для лучшего понимания алгоритма «Система позволяет печать билет для пассажира» была создана схема алгоритма, которая изображена на рисунке 24.

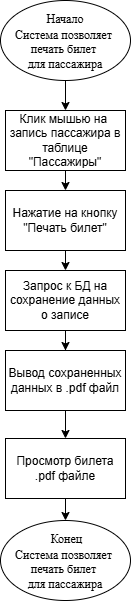


Рисунок 24 – аллгоритм «Система позволяет печать билет для пассажира»

Для лучшего понимания алгоритма «Система позволяет просматривать пассажиров за период времени на определенном маршруте» была создана схема алгоритма, которая изображена на рисунке 25.

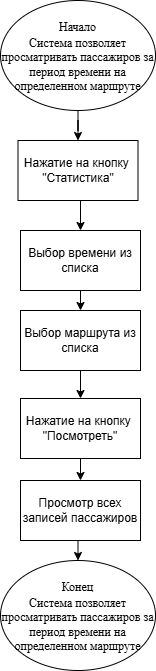


Рисунок 25 – алгоритм «Система позволяет просматривать пассажиров за период времени на определенном маршруте»

Для лучшего понимания алгоритма «Система позволяет отредактировать данные добавленного пассажира» была создана схема алгоритма, которая изображена на рисунке 26.



Рисунок 26 – алгоритм «Система позволяет отредактировать данные добавленного пассажира»

Для лучшего понимания алгоритма «Система позволяет удалить пассажира» была создана схема алгоритма, которая изображена на рисунке 27.

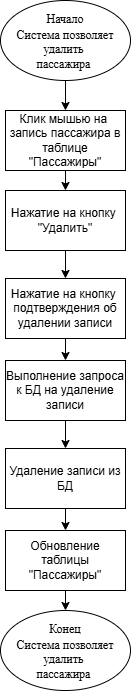


Рисунок 27 – алгоритм «Система позволяет удалить пассажира»

1. **Определение языка, структуры программы и требований к техническим средствам:**

Для реализации проекта будет использоваться язык программирования Python - универсальный язык программирования. Он подходит для решения широкого круга задач. Также у него простой синтаксис обширное количество фреймворков и библиотек.

Целевой СУБД является MySQL. MySQL также представляет собой систему управления базами данных с моделью данных реляционной базы данных. Эта особенность делает ее подходящей для различных предметных областей, подобно PostgreSQL.

Для создания графического визуала будет использоваться фреймворк Qt GUI с библиотекой PyQT5. Он предоставляет большое количество готовых виджетов и позволяет гибко настраивать интерфейс, поддерживая работу с событиями, сигналами и слотами. PyQt5 широко используется для создания настольных приложений и поддерживает интеграцию с базами данных, мультимедиа и сетевыми приложениями.

1. **Требования к техническим средствам:**

Приложение должно запускаться на персональных компьютерах с архитектурой х64, обладающих следующими минимальными характеристиками:

* Операционная система: Windows 10;
* Минимальное разрешение экрана 800x600.
* Видеокарта: с поддержкой DirectX 9.0c.
* Оперативная память: 2 Гб.
* СУБД: MySQL.

Программа должна запускаться со следующими требованиями системы обеспечения:

* Монитор.
* Клаивиатура и компьютерная мышь/тачпад.